Journal of Applied Data Sciences
Vol. 7, No. 1, January 2026, pp. 743-757
ISSN 2723-6471

Adaptive Test Model Enhancement Based on Salmon Salar Optimization and Partially Observable Markov Decision Process

Rujianto Eko Saputro[1], Fandy Setyo Utomo[2,*], Linda Perdana Wanti[3]

[1,2]Department of Computer Science, Universitas Amikom Purwokerto, Jl. Let. Jend. Pol. Soemarto, Purwokerto 53127, Indonesia

[3]Department of Cyber Security Engineering, Politeknik Negeri Cilacap, Jl. Dr. Soetomo No.1 Sidakaya, Cilacap 53212, Indonesia

(Received: August 15, 2025; Revised: October 15, 2025; Accepted: December 30, 2025; Available online: January 31, 2026)

Abstract

Cognitive Diagnosis Models (CDMs) in Computerized Adaptive Testing (CAT) are widely used to assess students' cognitive abilities; however, existing approaches face significant limitations. The Latent Trait Model often suffers from specification errors due to its complexity, the Diagnostic Classification Model encounters difficulties in integrating hierarchical structures, and Deep Learning Models demand substantial computational resources. To address these challenges, this study introduces Salmon Salar Optimization (SSO) to enhance CDM performance and integrates the Partially Observable Markov Decision Process (POMDP) to improve dynamic question selection. The proposed adaptive testing framework comprises three components: preprocessing, CDM, and a selection algorithm. Experimental results on the ASSISTments 2009-2010 dataset demonstrate that SSO outperforms representative baselines from both deep learning: Neural CD and Latent Trait Model: MIRT approaches. Using 5-fold cross-validation, the proposed model achieved superior predictive performance with 75.51% accuracy and an AUC of 0.8191, highlighting its robustness compared to existing state-of-the-art methods. Furthermore, adaptive test simulations reveal that the SSO- and POMDP-based model delivers superior outcomes, attaining 80.3% accuracy with a reward of 8.03 for 10-question exams and 79.8% accuracy with a reward of 11.97 for 15-question exams. These findings confirm the effectiveness of the proposed model in enhancing cognitive diagnosis and adaptive testing performance.

Keywords: Computerized Adaptive Testing, Cognitive Diagnostic Model, Greedy Strategy, Reinforcement Learning, Selection Algorithm

1. Introduction

The development of e-learning has changed the academic evaluation system, but still faces challenges in assessment flexibility that does not take into account differences in levels of understanding [1], learning styles, and students' thinking skills. Conventional examination systems often do not suit individual needs [2], so that a more adaptive and data-based academic evaluation is needed to reflect student competencies. Computerized Adaptive Testing (CAT) emerged as a solution in adaptive assessment [3]. The challenge in CAT is not only to adjust the difficulty level of the questions, but also to ensure that the questions selected are appropriate to the individual characteristics of the students, including previous performance patterns, learning style preferences, and level of question discrimination [4].

The Cognitive Diagnosis Model (CDM) in the CAT system is used to identify students' cognitive abilities and knowledge based on observation data [5], [6]. CDM is used in Computerized Adaptive Testing (CAT) to analyze student responses by considering item discrimination, difficulty level, and guessing factors. The three methods in CDM: Latent Trait Model, Diagnostic Classification Model, and Deep Learning Model have limitations, such as the high complexity of the Latent Trait Model that leads to specification errors, the challenge of integrating hierarchical structures in the Diagnostic Classification Model, and the expensive resource usage of the Deep Learning Model [7], [8]. The efficacy of Latent Trait Models, particularly within the framework of Item Response Theory (IRT), is frequently predicated on the axiom that latent traits adhere to a Gaussian distribution. However, empirical literature suggests that this constraint often induces model misspecification when applied to datasets exhibiting skewness or non-normality. Consequently, violating this distributional assumption may compromise parameter estimation accuracy and

***Corresponding author: Fandy Setyo Utomo (fandy_setyo_utomo@amikompurwokerto.ac.id)**
DOI: https://doi.org/10.47738/jads.v7i1.1065

attenuate the statistical robustness required in psychometric evaluations [9]. Furthermore, the Diagnostic Classification Model (DCM), although widely utilized for analyzing educational data, encounters notable limitations, particularly in its ability to incorporate hierarchical structures within the modeling framework. Recent advances in sequential hierarchical DCMs seek to capture the complexity of students' learning trajectories by accounting for multiple attributes and their temporal interactions. A study by [10] indicates that incorporating multi-level attributes can improve diagnostic precision; however, this enhancement also increases structural complexity, potentially elevating computational overhead and introducing classification inconsistencies due to the model's increased intricacy. In another domain, Deep Learning models, despite their strong capabilities for processing unstructured data, pose challenges in terms of resource requirements. They typically demand significant computational capacity and large training datasets, which can limit their feasibility for smaller educational institutions or research projects with constrained resources [11].

This study proposes the use of Salmon Salar Optimization (SSO), which is a metaheuristic-based optimization method that mimics the migration behavior of salmon in finding the best path [12] to improve the performance of CDM in identifying cognitive abilities. This research also integrates Reinforcement Learning (RL) techniques as a Selection Algorithm model within the CAT. This technique selects questions based on students' cognitive abilities and real-time performance in answering questions. Partially Observable Markov Decision Process (POMDP) is a selection algorithm model used to optimize question selection dynamically, so that the system not only selects questions based on the student's answer history, but is also able to learn from student interactions and adjust its strategy [13].

This study integrates the CDM model and the Selection Algorithm into a CAT system. The CDM model is based on SSO to optimize cognitive ability prediction, while the RL technique as a Selection Algorithm is capable of selecting questions based on students' cognitive abilities as SSO output and real-time student performance in answering questions. The results of this model are expected to improve the accuracy of academic evaluation, provide a more personalized exam experience, and ensure that students receive assessments that are appropriate to their abilities.

2. Literature Review

2.1. Computerized Adaptive Testing

Computerized Adaptive Testing (CAT) is an assessment approach that dynamically adjusts item difficulty in real-time based on the examinee's responses, continuously estimating individual ability to select questions that best measure proficiency. Unlike traditional fixed tests, CAT personalizes the assessment for each test-taker, enhancing both measurement precision and the testing experience [14]. The CAT workflow functions as an adaptive cycle that starts with a medium-level question, then proceeds through repeated stages of item selection, response analysis, and ability estimation. Each updated ability estimate guides the choice of the next item to optimize measurement accuracy. The process continues until specific stopping conditions, such as target precision or maximum test length, are satisfied, after which the final performance score is generated. This iterative system provides an efficient and individualized testing process [15].

The primary components of a CAT system can be categorized as follows: (1) The item bank: is a collection of diverse test items with defined parameters, such as difficulty and discrimination [16], that allows CAT to dynamically select questions matching the test taker's ability [17]. A sufficiently large and varied bank ensures robust assessment across ability levels [18]; (2) The adaptive algorithm: in CAT selects items based on examinee responses, starting with an initial item to estimate ability. Subsequent questions are adjusted in difficulty—harder for correct answers and easier for incorrect ones [19]—ensuring appropriate challenge, engagement, and reduced frustration [20]; (3) Ability Estimation Method: In CAT, ability estimation is updated after each response using methods like Maximum Likelihood Estimation or Bayesian estimation [21]. This iterative process refines the examinee's trait level, guiding item selection and improving assessment accuracy [22]; and (4) User Interface and Accessibility: Since CAT is delivered through computers, a well-designed user interface is essential to ensure smooth navigation. An intuitive interface minimizes cognitive load, enabling test takers to concentrate on the content and thereby enhancing both performance and satisfaction [23]. Our research focus is to optimize the adaptive algorithm in CAT.

2.2. Salmon Salar Optimization

Research conducted by [12] introduces SSO, a metaheuristic method inspired by the social behavior of salmon: food search, migration to spawning habitats, and danger avoidance, to solve complex optimization problems. Employing Salmon Salar as an algorithmic model offers several notable benefits. First, it provides an effective balance between exploration and exploitation by simulating salmon's collective behaviors in foraging, migration, and risk avoidance, which allows the algorithm to search extensively while simultaneously refining superior solutions. Second, the inherent social structure of salmon supports position updating via Gaussian distribution, thereby maintaining adaptability and robustness in solving high-dimensional and complex optimization tasks. Third, the incorporation of danger-avoidance mechanisms reduces the risk of premature convergence to local optima and facilitates progress toward global optima. Lastly, its practical utility has been validated in the optimization of deep-sea probe design for unconventional oil exploration, where the method achieves accurate outcomes despite multiple variables and constraints. In this study, we propose SSO as a component of the adaptive test model to have better performance than existing approaches.

3. Methodology

This section consists of two sub-chapters: data collection and proposed methods. The data collection section explains the sources, attributes, and conditions of the dataset used in this study, while the proposed methods section describes the new method proposed for the adaptive test model.

3.1. Data Collection

In this study, we used the ASSISTments 2009-2010 dataset, which is publicly accessible at https://sites.google.com/site/assistmentsdata/home/2009-2010-assistment-data. There are several versions of the ASSISTments 2009-2010 dataset, but we used the skill_builder_data_corrected_collapsed.csv dataset for the experiment because the data already contains one row per student-problem. There are 31 attributes and 346860 rows of data in the dataset with target variable: 'correct' -- true/false answer (1 or 0). The Skill Builder problem sets are designed with several key features. Each question focuses on a specific skill, and it may be tagged with multiple skill labels. To successfully complete the assignment, students must answer three consecutive questions correctly. If a student uses the tutoring options, such as hints or step-by-step guidance, the question is automatically marked as incorrect. Additionally, students receive immediate feedback on whether their answer is correct. In cases where a student struggles to solve the problem independently, the final hint will provide the correct answer.

Not all attributes were used in the experiment, but we only used 12 main attributes: 'user_id,' 'problem_id,' 'correct,' 'ms_first_response,' 'attempt_count,' 'hint_count,' 'assignment_id,' 'sequence_id,' 'assistment_id,' 'answer_type,' 'first_action,' and 'skill_id.' Each attribute in the dataset provides key information about student interactions with the problem set. 'user_id' identifies the student, while 'problem_id' tracks the specific question. 'correct' indicates whether the answer is correct (1) or incorrect (0). 'ms_first_response' records the time taken to respond, and 'attempt_count' captures the number of attempts made. 'hint_count' shows how many hints were used. 'assignment_id' identifies the specific assignment, and 'sequence_id' tracks the question order. 'assistment_id' connects questions to a set of problems, while 'answer_type' denotes the format of the answer. 'first_action' tracks the student's first move, and 'skill_id' identifies the skill assessed by the question.

3.2. Proposed Method

Figure 1 shows the three main components in developing an adaptive exam model: the preprocessing component, the CDM, and the Selection Algorithm. We propose Salmon Salar Optimization as the CDM component. Furthermore, we also propose POMDP-based Reinforcement Learning as the selection algorithm component.

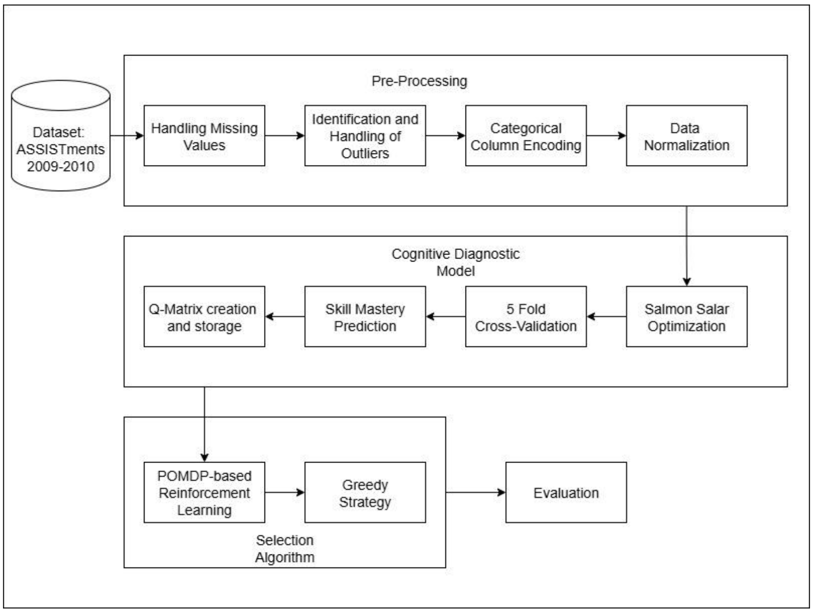


Figure 1. Proposed Method

An explanation of each component of the adaptive exam model and its working procedures is provided in sections 3.3 to 3.6. Section 3.3 describes the pre-processing components, Section 3.4 presents the Cognitive Diagnostic Model components, Section 3.5 represents the selection algorithm components, and the final section explains the evaluation techniques applied to assess the performance of the selection algorithm components.

3.3. Pre-processing Component

Based on figure 1, there are several stages in the pre-processing component: handling missing values, identifying and handling outliers, handling categorical columns, and data normalization. First, identify and handle missing values by eliminating data rows with missing values in the columns 'user_id', 'problem_id', 'correct', 'ms_first_response', 'attempt_count', 'hint_count', 'assignment_id', 'sequence_id', 'assistment_id', 'answer_type', and 'first_action' to ensure clean data.

After handling missing values, the next step is to identify and handle outliers using Z-score on the numeric columns: 'ms_first_response', 'attempt_count', 'hint_count', 'assignment_id', 'sequence_id', 'assistment_id.' The numeric data in these columns is checked and removed if it has an outlier value based on the Z-score threshold of more than 3. The Z-score is a statistical metric used to detect outliers by showing how far a data point deviates, in terms of standard deviations, from the dataset's mean. This approach assumes that the data follows a roughly normal distribution. Data points exceeding a predefined Z-score threshold are considered outliers [24], [25].

The next stage is to handle categorical columns. The categorical 'answer_type' column is converted to a numeric representation using Label Encoding. Next, data normalization is performed on the columns 'ms_first_response', 'attempt_count', 'hint_count', 'assignment_id', 'sequence_id', and 'assistment_id' using Z-score normalization. Z-score normalization, or standardization, is a method applied to rescale data to a common standard, facilitating comparisons across datasets, especially when the original data have differing units or scales. This process transforms the data so that the resulting distribution has a mean of 0 and a standard deviation of 1, allowing for uniform and standardized comparisons [26], [27], [28].

3.4. Cognitive Diagnostic Model Component

After the pre-processing stage, the CDM component processes the data. Cognitive Diagnostic Models (CDMs), also referred to as Diagnostic Classification Models (DCMs), constitute an advanced class of models in educational assessment designed to assess and quantify learners' proficiency in specific skills or attributes. In contrast to conventional measurement approaches that produce general performance summaries, CDMs offer detailed insights into individual learners' strengths and weaknesses across distinct cognitive skills [29], [30].

There are several stages in the CDM component: Salmon Salar Optimization (SSO), 5-fold cross-validation, skill mastery prediction, and Q-matrix creation. In this study, we propose SSO to identify cognitive abilities. The following is the working procedure of the SSO algorithm [12].

Randomly initialize the population of salmons. Identify the Leader Salmon (the individual with the best fitness). Update the positions of salmons using the Gaussian-based equations. Update the position of the Leader Salmon based on the best solutions in the population. Repeat the iteration until the maximum number of generations is reached. Output the optimal solution (Leader Salmon) as the final result of the optimization. Table 1 describe mapping between Salmon Salar's work procedures and the representation of the Cognitive Diagnostic Model in this study.

Table 1. SSO Procedure Mapping with CDM Data Representation

Stage	SSO Stage	Conceptual Description	CDM Data Representation
1	Population Initialization	SSO begins by randomly generating an initial population of salmon within the search space.	Each salmon position is a single long vector that combines all the parameters of the CDM model: each student's ability, each question's difficulty, and the feature weights.
2	Fitness Evaluation	Each salmon position is evaluated for quality using an objective function (fitness function).	Calculating Negative Log-Likelihood. The smaller the value (the more negative), the more accurate the parameters (ability, difficulty, weights) are in predicting student responses (correct or incorrect) using the Sigmoid function.
3	Salmon Leader Determination	The salmon with the best fitness value in the population is selected as the Leader.	The currently best combination of CDM parameters that yields the lowest prediction error on the training data is stored in a variable. This is the best tentative estimate of student ability and problem characteristics.
4	Exploration & Exploitation	Salmon updates its position by moving closer to the Leader but still performs random exploration using a Gaussian distribution to avoid local optimum traps.	create new variations of the CDM parameters based on the average values of the current salmon and Leader positions to simulate a more precise parameter search
5	Best Candidate Selection	Compare the current position of the salmon with the newly generated candidate positions. The position with the best fitness value will be selected.	Comparing the Log-Likelihood of the old CDM parameters with the modified parameters. If the new parameters predict student data more accurately, they are retained in place of the old ones.
6	Global Update & Termination	The process repeats until the maximum iteration is reached. The final leader is considered the global optimal solution.	After the iterations are complete, the Leader variable is further broken down into the final variables: *student_ability_global*, *global_difficulty*, and *global_weights*. These variables are used for prediction on the validation data.

The next phase uses 5-fold cross-validation. During each fold, Salmon Salar Optimization (SSO) optimizes the parameters of the CDM on the training data. SSO searches for optimal parameters such as user ability and problem difficulty, which the model then uses to make predictions on the validation data. After optimization, the model evaluates the validation data using metrics like accuracy and AUC. This process repeats across five folds, ensuring diverse data subsets are used for both training and testing. The results from all folds provide a comprehensive evaluation of the optimized model's performance. SSO adjusts the parameters for each fold, minimizing overfitting, while 5-fold cross-validation ensures more reliable generalization to unseen data. Ultimately, the optimal parameters assess the model's ability to predict skill mastery using the validation data.

Once the parameters are optimized, skill mastery is calculated using the logit formula followed by the sigmoid function. Predictions for skill mastery are calculated using the logit equation (1).

$$logit = ability[user_{idx}] - difficulty[problem_{idx}] + weighted\ sum\ of\ features \tag{1}$$

ISSN 2723-6471

ability[user$_{idx}$] refers to the user's proficiency on a specific problem, *difficulty[problem$_{idx}$]* represents the difficulty level of the given problem. Then, the weighted sum of features encompasses additional weights calculated based on features such as response time, number of attempts, hint usage, problem type, and the user's first action. Next, the prediction of skill mastery is calculated using the sigmoid function (2).

$$Skill\ Mastery = \sigma(logit) = \frac{1}{1+e^{-logit}} \tag{2}$$

where σ is the sigmoid function that converts logit values into probabilities between 0 and 1. This method transforms the logit into a probability, representing the likelihood of a user mastering a specific skill. The model's performance is then evaluated by calculating the skill mastery prediction for each problem, yielding probabilities that indicate the likelihood of the user mastering a particular skill. Finally, the predicted skill mastery results are stored in a CSV file for further analysis, allowing for subsequent evaluations and comparisons. To ensure reproducibility and accessibility of these findings, the output CSV files have been deposited in a public repository and can be accessed at https://github.com/fandysetyoutomo-dev/predict_skill_mastery.git.

Next, enter the creation stage of Q-Matrix. The fundamental principle behind CDMs is mapping responses in assessment items directly to latent cognitive attributes defined in advance, often organized in a Q-matrix. This Q-matrix delineates the relationship between test items and underlying skills or attributes, enabling researchers to identify which specific competencies a learner has mastered and where they need improvement [31], [32], [33], [34]. In the context of this research, a Q-Matrix is a matrix that describes the relationship between a problem and the skills required to solve it. This process involves mapping problem IDs to relevant skill IDs.

3.5. Selection Algorithm Component

After obtaining the data from the skill mastery prediction and Q-matrix results, the data then acts as input to the selection algorithm component. The selection algorithm constitutes a central mechanism that determines the choice of items (test questions) according to a test-taker's ability and performance. Its primary objective is to personalize assessments, thereby improving the testing experience and increasing the precision of ability estimates. The selection algorithm is pivotal to Computerized Adaptive Testing (CAT), as it enhances test administration by optimizing both the efficiency and accuracy of assessments [35], [36]. An effectively executed selection algorithm enables the dynamic adjustment of item difficulty, ensuring that test-takers are presented with items of appropriate challenge —neither too easy nor too difficult —thus sustaining motivation and engagement [37], [38].

In this study, we propose the Reinforcement Learning: Partially Observable Markov Decision Process (POMDP) model as a selection algorithm. The POMDP is a mathematical framework designed for decision-making in environments where an agent has partial information about the current system state. POMDPs expand on the basic Markov Decision Processes (MDPs) by incorporating observations and beliefs, allowing for the handling of situations where the state cannot be fully observed due to uncertainty [39], [40]. There are several key components in POMDP [41]:

The states (S) represent the selection of a specific test item (question) from the item bank to be presented to the student. The actions (A) are the decision taken by the agent (policy network) at each time step *t* to choose which Skill ID to test or train the next student. Observations (Z) are the student's response to the selected question (e.g., correct (1) or incorrect (0)). The transition function (T) specifies the probability of a student's knowledge state changing after interacting with a question. The reward function (R) outlines the immediate feedback received by the system, defined as a positive value for a correct answer (accuracy) or information gain regarding the student's true ability. The observation function (O) are the probability of students answering correctly based on the current state and the difficulty level of the question using the Sigmoid function. Finally, the initial belief (b0) describes the initial estimate of the probability of student mastery for each skill before the session begins [42], [43].

The relationship between Q-matrix, Predict Skill Mastery, and POMDP is integral in optimizing assessment models. The Q-matrix provides information about the skills measured by each problem, which is used in POMDP to update the belief state. After each question is answered, the belief about the user's skills is updated, with the Q-matrix mapping the relevant skills to the problem. The Q-matrix also plays a critical role in predicting skill mastery, as it determines the skills associated with a given question. The prediction is based on the optimized user ability, problem difficulty, and additional feature weights. If a question assesses an underdeveloped skill, the prediction will reflect the user's lack of mastery in that area. Lastly, in POMDP, the belief state is continually updated after each question, and skill mastery predictions represent the user's proficiency based on the problems they encounter. POMDP enables the system to select questions that better explore undeveloped skills, thus enhancing the selection process and updating the belief state with more precise information.

Furthermore, within the framework of Partially Observable Markov Decision Processes (POMDPs), the greedy strategy plays a crucial role in decision-making under uncertainty. A POMDP is a mathematical model used to address decision-making scenarios where the agent lacks complete information about the environment's state, often requiring decisions based on belief states rather than fully observable states. Greedy algorithms streamline the decision-making process by selecting the action with the highest immediate expected reward, thereby reducing computational complexity in domains where evaluating all possible states and actions is computationally expensive [44], [45]. In the context of this research, the Greedy Strategy was used in the post-training evaluation. Question selection was carried out greedily, namely selecting questions based on the highest probability from the POMDP policy network. Equation (3) shows the greedy strategy formula implemented in this study.

$$\sigma_{greedy} = \frac{\arg\max P(\sigma|\theta)}{\sigma} \quad (3)$$

σ_{greedy} is the action (or problem) chosen by the agent, $\frac{arg\,max}{\sigma}$ denotes selecting the action σ that maximizes the probability $P(\sigma|\Theta)$, while $P(\sigma|\Theta)$ is the predicted probability of a user mastering the skill associated with action σ, given the model parameters Θ, which include user ability, problem difficulty, and additional feature weights.

3.6. Evaluation

The evaluation of POMDP and greedy strategy in this research focuses on measuring accuracy and reward based on the decisions made by the agent. In POMDP, the agent updates its belief state after each question, allowing for adaptive selection of problems that best reveal the user's strengths and weaknesses. The proportion of correct predictions determines accuracy, while a reward is given for each correct answer. In the greedy strategy, the agent selects the question with the highest predicted probability of success, based on the current model, and the accuracy and reward are similarly calculated. The greedy approach simplifies decision-making by focusing on the most likely correct answer, while POMDP offers a more flexible, belief-based approach. Both strategies evaluate how well the model predicts skill mastery and guide question selection based on user performance.

4. Results and Discussion

This chapter comprises several sections: Section 4.1 presents the results of pre-processing experiments, Section 4.2 discusses the findings of the Cognitive Diagnostic Model, and Section 4.3 describes the outcomes of the Selection Algorithm experiments. Finally, Section 4.4 discusses the performance comparison between the proposed adaptive test model and previous test models.

4.1. Pre-processing and Cognitive Diagnostic Model Results

The missing value identification results show that the dataset contains no missing data. Additionally, table 2 presents the results of outlier identification across several columns.

Table 2. Outliers Identification Results

No.	Column Name	Number of rows deleted
1.	ms_first_response	630
2.	attempt_count	436
3.	hint_count	5067
4.	assignment_id	5542
5.	sequence_id	36
6.	assistment_id	4003

ISSN 2723-6471

Based on table 2, the final number of rows after outlier elimination: 331146. Next, figure 2 shows the SSO loss history across CDM Folds. This figure illustrates the evolution of the Negative Log-Likelihood (NLL) loss across five cross-validation folds during iterative optimization. The x-axis represents the number of iterations, while the y-axis denotes the corresponding Negative Log-likelihood loss values, which serve as a measure of model performance. Across all folds, the loss exhibits a steep decline during the initial iterations, indicating that the model rapidly captures fundamental patterns in the data. Subsequently, the rate of decline diminishes, and the curves gradually converge toward stable values, reflecting the model's progression toward convergence. Notably, differences among folds are observed: Fold 3 consistently achieves the lowest loss values, suggesting superior model performance under this partition of the data, while Fold 4 yields comparatively higher loss values, pointing to weaker predictive accuracy in this scenario. This variation is likely due to the stochastic characteristics of random data partitioning in cross-validation. In particular, Fold four may include a larger number of sparse user–item interactions or "cold-start" cases in which the model has insufficient historical information to generate reliable predictions. In contrast, Fold three appears to comprise a more homogeneous subset with denser interaction histories, enabling faster and more stable model convergence. The remaining folds demonstrate intermediate performance with relatively minor variations. Overall, the results indicate that the SSO optimization process effectively reduces loss across folds, with convergence reached within a limited number of iterations, though variability across folds highlights the influence of data partitioning on model outcomes.

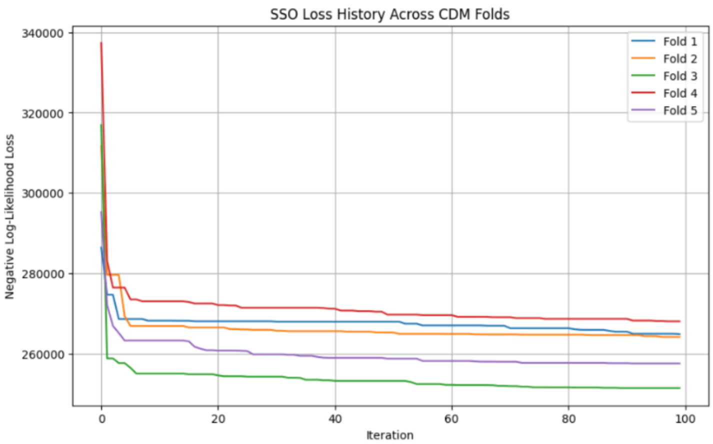


Figure 2. SSO Loss History

Next, figure 3 describes the results of the SSO evaluation using accuracy and AUC in each fold.

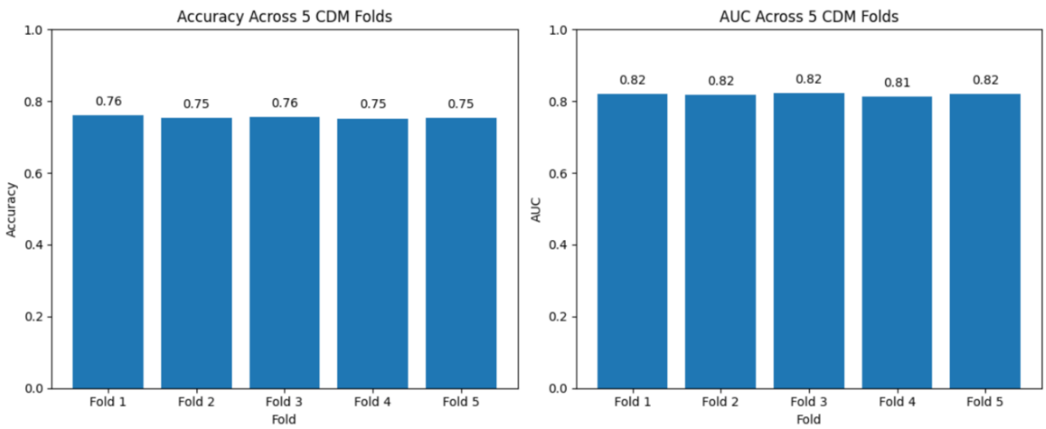


Figure 3. Accuracy and AUC Evaluation on SSO

Based on figure 3, the results obtained were average accuracy in 5 folds: 0.7551 and average AUC in 5 folds: 0.8191. The presented bar charts depict the performance of the model across five folds of cross-validation in terms of accuracy and the Area Under the Curve (AUC). As shown in the left panel, the accuracy values consistently range between 0.75 and 0.76 across all folds, indicating that the model correctly predicts approximately three-quarters of the instances regardless of the data partition. The minimal variation among folds demonstrates the stability and robustness of the model's predictive capability. The right panel illustrates the AUC values, which lie between 0.81 and 0.82 across folds. These results suggest that the model possesses a strong discriminatory ability in distinguishing between positive and negative classes, as AUC values above 0.80 are generally regarded as indicative of good classification performance. Taken together, the results confirm that the model exhibits consistent accuracy and reliable discrimination across multiple folds, thereby underscoring its generalizability and robustness in handling different subsets of the data. After evaluating and validating SSO in the 5-fold cross-validation phase, the model predicted skill mastery. Table 3 describes an example of the output from the calculation results of the mastery skill prediction.

Table 3. Skill Mastery Prediction Output Example

user_id	problem_id	skill_mastery	Label
64525	51424	0.932875	1
64525	51435	0.966380	1
70695	51434	0.990699	1
70695	51473	0.755382	1
70699	51430	0.993342	1

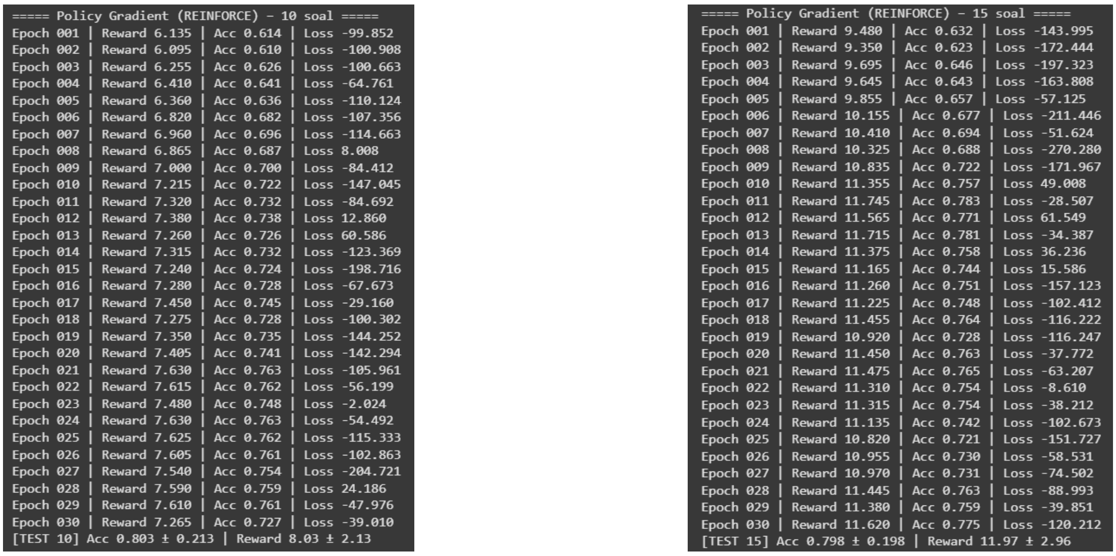
Based on table 3, the predict_skill_mastery.csv output presents the predicted skill mastery for individual users on specific problems. Each row corresponds to a unique combination of user_id and problem_id, where skill_mastery represents the model's predicted probability that the user has mastered the skill required to solve that problem. Values close to 1 indicate a high likelihood of mastery, while values near 0 suggest limited proficiency. The Label column provides the ground-truth response, with 1 indicating a correct answer and 0 indicating an incorrect response. This dataset enables a direct comparison between predicted skill mastery and actual performance, facilitating the evaluation of model accuracy and discriminative ability. Furthermore, the information can be employed to analyze individual learning progress and inform adaptive testing strategies, such as selecting subsequent problems based on estimated proficiency, thereby optimizing the assessment process.

The next step is to create a Q-matrix. Based on the experimental results, 8074 problem_ids have no skill mapping, and the Q-matrix has dimensions of 24762 x 122. The Q-matrix output indicates that the file Q_matrix.npy has dimensions of 24762 × 122, representing 24762 unique problems and 122 distinct skills. Each row corresponds to a specific problem, while each column represents a particular skill, with a value of 1 indicating that the problem assesses the corresponding skill and 0 indicating no association. The warning that 8074 problem_ids lack a skill mapping signifies that these problems do not have associated skill information, and their corresponding rows in the Q-matrix remain zero. Despite this, all problems are retained in the matrix to maintain consistency with the dataset. The Q-matrix serves a critical function in linking problems to relevant skills, supporting the calculation of predicted skill mastery, updating belief states within the POMDP framework, and guiding adaptive or reinforcement learning-based problem selection. Problems without skill mappings, however, do not contribute to the assessment of specific skills, highlighting potential gaps in the dataset that may affect the precision of skill mastery predictions.

4.2. Selection Algorithm Results

Next, the POMDP and Greedy strategy evaluations were conducted using 10 and 15 exam simulation questions, respectively. Figure 4(a) shows the evaluation results for the 10-question simulation, and figure 4(b) describes the evaluation results for the 15-question simulation. Based on figure 4(a) it can be explained that the training output for the 10-question setting demonstrates the learning progression of the Policy Gradient (REINFORCE) algorithm within the POMDP framework. Over 30 epochs, the model progressively improves its ability to select questions that align with users' skill levels, as indicated by the increase in reward from approximately 6.14 to 7.63 and accuracy from 0.614 to 0.763. The loss fluctuates throughout training, reflecting the stochastic updates inherent in policy gradient methods. The final evaluation, with an average accuracy of 0.803 ± 0.213 and a reward of 8.03 ± 2.13, suggests that the learned policy effectively guides question selection in a manner that maximizes correct responses and reflects predicted skill mastery, demonstrating the model's ability to adaptively optimize problem selection for a smaller question set. It is acknowledged that the standard deviation for accuracy is relatively high (± 0.213). This variance is primarily attributed to the heterogeneity of the student population in the ASSISTments dataset, particularly the presence of 'cold-start' students with sparse interaction histories. While the model achieves high accuracy for students with sufficient data, the predictive performance naturally varies for users with limited prior information. Despite this, the mean accuracy demonstrates the overall effectiveness of the proposed SSO-POMDP framework.

Meanwhile, based on figure 4(b) it can be described that for the 15-question setting, the Policy Gradient (REINFORCE) algorithm exhibits a similar learning trend, with rewards increasing from 9.48 to 11.62 and accuracy improving from 0.632 to 0.775 over 30 epochs. Despite fluctuations in the loss values, the model effectively learns a policy that prioritizes questions most likely to be answered correctly, optimizing the exploration of users' skill mastery. The final test-phase results, with an average accuracy of 0.798 ± 0.198 and a reward of 11.97 ± 2.96, indicate that the policy maintains strong performance across a larger question set. These outcomes confirm that the POMDP framework, combined with the greedy strategy, successfully adapts question selection to the users' skill levels, enhancing both the efficiency and accuracy of skill mastery prediction in longer assessment sessions.



```
===== Policy Gradient (REINFORCE) - 10 soal =====
Epoch 001 | Reward 6.135 | Acc 0.614 | Loss -99.852
Epoch 002 | Reward 6.095 | Acc 0.610 | Loss -100.908
Epoch 003 | Reward 6.255 | Acc 0.626 | Loss -100.663
Epoch 004 | Reward 6.410 | Acc 0.641 | Loss -64.761
Epoch 005 | Reward 6.360 | Acc 0.636 | Loss -110.124
Epoch 006 | Reward 6.820 | Acc 0.682 | Loss -107.356
Epoch 007 | Reward 6.960 | Acc 0.696 | Loss -114.663
Epoch 008 | Reward 6.865 | Acc 0.687 | Loss 8.008
Epoch 009 | Reward 7.000 | Acc 0.700 | Loss -84.412
Epoch 010 | Reward 7.215 | Acc 0.722 | Loss -147.045
Epoch 011 | Reward 7.320 | Acc 0.732 | Loss -84.692
Epoch 012 | Reward 7.380 | Acc 0.738 | Loss 12.860
Epoch 013 | Reward 7.260 | Acc 0.726 | Loss 60.586
Epoch 014 | Reward 7.315 | Acc 0.732 | Loss -123.369
Epoch 015 | Reward 7.240 | Acc 0.724 | Loss -198.716
Epoch 016 | Reward 7.280 | Acc 0.728 | Loss -67.673
Epoch 017 | Reward 7.450 | Acc 0.745 | Loss -29.160
Epoch 018 | Reward 7.275 | Acc 0.728 | Loss -100.302
Epoch 019 | Reward 7.350 | Acc 0.735 | Loss -144.252
Epoch 020 | Reward 7.405 | Acc 0.741 | Loss -142.294
Epoch 021 | Reward 7.630 | Acc 0.763 | Loss -105.961
Epoch 022 | Reward 7.615 | Acc 0.762 | Loss -56.199
Epoch 023 | Reward 7.480 | Acc 0.748 | Loss -2.024
Epoch 024 | Reward 7.630 | Acc 0.763 | Loss -54.492
Epoch 025 | Reward 7.625 | Acc 0.762 | Loss -115.333
Epoch 026 | Reward 7.605 | Acc 0.761 | Loss -102.863
Epoch 027 | Reward 7.540 | Acc 0.754 | Loss -204.721
Epoch 028 | Reward 7.590 | Acc 0.759 | Loss 24.186
Epoch 029 | Reward 7.610 | Acc 0.761 | Loss -47.976
Epoch 030 | Reward 7.265 | Acc 0.727 | Loss -39.010
[TEST 10] Acc 0.803 ± 0.213 | Reward 8.03 ± 2.13
```

(a)

```
===== Policy Gradient (REINFORCE) - 15 soal =====
Epoch 001 | Reward 9.480 | Acc 0.632 | Loss -143.995
Epoch 002 | Reward 9.350 | Acc 0.623 | Loss -172.444
Epoch 003 | Reward 9.695 | Acc 0.646 | Loss -197.323
Epoch 004 | Reward 9.645 | Acc 0.643 | Loss -163.808
Epoch 005 | Reward 9.855 | Acc 0.657 | Loss -57.125
Epoch 006 | Reward 10.155 | Acc 0.677 | Loss -211.446
Epoch 007 | Reward 10.410 | Acc 0.694 | Loss -51.624
Epoch 008 | Reward 10.325 | Acc 0.688 | Loss -270.280
Epoch 009 | Reward 10.835 | Acc 0.722 | Loss -171.967
Epoch 010 | Reward 11.355 | Acc 0.757 | Loss 49.008
Epoch 011 | Reward 11.745 | Acc 0.783 | Loss -28.507
Epoch 012 | Reward 11.565 | Acc 0.771 | Loss 61.549
Epoch 013 | Reward 11.715 | Acc 0.781 | Loss -34.387
Epoch 014 | Reward 11.375 | Acc 0.758 | Loss 36.236
Epoch 015 | Reward 11.165 | Acc 0.744 | Loss 15.586
Epoch 016 | Reward 11.260 | Acc 0.751 | Loss -157.123
Epoch 017 | Reward 11.225 | Acc 0.748 | Loss -102.412
Epoch 018 | Reward 11.455 | Acc 0.764 | Loss -116.222
Epoch 019 | Reward 10.920 | Acc 0.728 | Loss -116.247
Epoch 020 | Reward 11.450 | Acc 0.763 | Loss -37.772
Epoch 021 | Reward 11.475 | Acc 0.765 | Loss -63.207
Epoch 022 | Reward 11.310 | Acc 0.754 | Loss -8.610
Epoch 023 | Reward 11.315 | Acc 0.754 | Loss -38.212
Epoch 024 | Reward 11.135 | Acc 0.742 | Loss -102.673
Epoch 025 | Reward 10.820 | Acc 0.721 | Loss -151.727
Epoch 026 | Reward 10.955 | Acc 0.730 | Loss -58.531
Epoch 027 | Reward 10.970 | Acc 0.731 | Loss -74.502
Epoch 028 | Reward 11.445 | Acc 0.763 | Loss -88.993
Epoch 029 | Reward 11.380 | Acc 0.759 | Loss -39.851
Epoch 030 | Reward 11.620 | Acc 0.775 | Loss -120.212
[TEST 15] Acc 0.798 ± 0.198 | Reward 11.97 ± 2.96
```

(b)

Figure 4. (a) 10 Exam Questions Simulation; (b) 15 Exam Questions Simulation

4.3. Discussion

In this study, we conducted a performance comparison between the proposed models SSO and POMDP, with Neural CD [46] and POMDP, and Multidimensional Item Response Theory (MIRT) [47] and POMDP. Table 4 shows the comparison of accuracy and AUC average performance using 5 folds cross-validation on three models.

Table 4. Cognitive Diagnostic Model Performance Comparison

	SSO	Neural CD	MIRT
Accuracy	75.51%	73.73%	65.87%
AUC	0.8191	0.7472	0.6963

Based on table 4, it can be concluded that SSO has better performance than the other two models in terms of accuracy and AUC scores. Next, table 5 describes the results of the performance comparison of the three CDM models in the POMDP environment with 10 questions.

Table 5. Adaptive Test Models Comparison (Q=10)

	SSO+POMDP	Neural CD+POMDP	MIRT+POMDP
Accuracy	80.3%	78.4%	60.1%
Reward	8.03	7.84	6.01

Based on table 5, it can be concluded that SSO+POMDP has better performance than the other two models in terms of accuracy and Reward scores. Furthermore, table 6 describes the results of the performance comparison of the three CDM models in the POMDP environment with 15 questions.

Table 6. Adaptive Test Models Comparison (Q=15)

	SSO+POMDP	Neural CD+POMDP	MIRT+POMDP
Accuracy	79.8%	73.1%	60.3%
Reward	11.97	10.97	9.05

Similar to the previous simulation, table 6 shows that SSO+POMDP outperforms the other two models in terms of accuracy and reward scores.

5. Conclusion

Our research successfully built a model and improved the performance of the adaptive test model through the proposed method using Salmon Salar Optimization (SSO) and Partially Observable Markov Decision Process (POMDP). The experimental results showed that the performance of SSO on the CDM component for cognitive ability prediction had the highest results compared to the other two models Neural CD and MIRT with an accuracy score of 75.51% and an AUC of 0.8191. Furthermore, in adaptive exam simulations using 10 and 15 questions, the proposed SSO and POMDP-based adaptive exam model had the best performance compared to the other two adaptive exam models. In the exam simulation with 10 questions, SSO and POMDP obtained an accuracy score of 80.3% and a reward of 8.03. Meanwhile, in the exam simulation with 15 questions, SSO and POMDP obtained an accuracy score of 79.8% and a reward of 11.97. Based on the experiments that have been conducted, there is still potential to improve the performance of the adaptive exam model using other approaches or techniques.

Despite the encouraging findings, this study presents several limitations that warrant consideration. First, with respect to scalability, the SSO metaheuristic may incur substantial computational overhead when deployed on very large item banks or in environments with simultaneous real-time users, indicating a need for further refinement before large-scale implementation. Second, in terms of interpretability, although CDMs yield valuable diagnostic information, their integration with a POMDP framework introduces additional complexity in the policy decision process, making it less transparent than traditional rule-based approaches and potentially challenging for educators seeking to understand the rationale behind each item selection. Lastly, the results are constrained by dataset dependency, as validation was conducted solely on the ASSISTments 2009–2010 dataset. Future studies should examine the generalizability of the proposed framework across multiple educational datasets and subject areas.

6. Declarations

6.1. Author Contributions

Conceptualization: R.E.S., F.S.U., and L.P.W.; Methodology: F.S.U.; Software: R.E.S.; Validation: R.E.S., F.S.U., and L.P.W.; Formal Analysis: R.E.S., F.S.U., and L.P.W.; Investigation: R.E.S.; Resources: F.S.U.; Data Curation: F.S.U.; Writing Original Draft Preparation: R.E.S., F.S.U., and L.P.W.; Writing Review and Editing: F.S.U., R.E.S., and L.P.W.; Visualization: R.E.S. All authors have read and agreed to the published version of the manuscript.

6.2. Data Availability Statement

This study utilized publicly accessible datasets, which can be obtained from: https://sites.google.com/site/assistmentsdata/home/2009-2010-assistment-data. Furthermore, the prediction outputs produced in this research are hosted in an open-access repository at: https://github.com/fandysetyoutomo-dev/predict_skill_mastery.git

ISSN 2723-6471

6.3. Funding

We sincerely acknowledge the Ministry of Higher Education, Science, and Technology of Indonesia for its financial support of this research. Such assistance has been invaluable in facilitating the research process and manuscript preparation, enabling us to achieve the best possible outcomes.

6.4. Institutional Review Board Statement

Not applicable.

6.5. Informed Consent Statement

Not applicable.

6.6. Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] H. Waheed, S. U. Hassan, N. R. Aljohani, J. Hardman, S. Alelyani, and R. Nawaz, "Predicting academic performance of students from VLE big data using deep learning models," *Comput. Human Behav.*, vol. 104, no. November 2019, pp. 1-19, 2020, doi: 10.1016/j.chb.2019.106189.

[2] I. A. M. S. Widiastuti, I. B. N. Mantra, H. Sukoco, and M. H. Santosa, "Online assessment strategies to enhance students' competence and their implementational challenges," *JEES (Journal English Educ. Soc.*, vol. 6, no. 2, pp.1-12, Oct. 2021, doi: 10.21070/jees.v6i2.1378.

[3] M. Adnan, A. Habib, J. Ashraf, S. Mussadiq, A. . Raza, and M. Abid, "Predicting at-Risk Students at Different Percentages of Course Length for Early Intervention Using Machine Learning Models," *IEEE Access,* vol. 9, no. 1, pp. 7519–7539, 2021, doi: 10.1109/ACCESS.2021.3049446.

[4] I. Ristić, M. Runić-Ristić, T. Savić Tot, V. Tot, and M. Bajac, "The Effects and Effectiveness of An Adaptive E-Learning System on The Learning Process and Performance of Students," *Int. J. Cogn. Res. Sci. Eng. Educ.*, vol. 11, no. 1, pp. 77–92, Apr. 2023, doi: 10.23947/2334-8496-2023-11-1-77-92.

[5] X. Wu, T. Xu, and R. Wu, "Research on personalized assessment Information of students' mathematical competency based on cognitive diagnosis," *Think. Ski. Creat.,* vol. 53, no.1, pp. 1-16, 2024, doi: 10.1016/j.tsc.2024.101596.

[6] Z. Shang, E. A. Erosheva, and G. Xu, "Partial-mastery cognitive diagnosis models," *Ann. Appl. Stat.*, vol. 15, no. 3, pp. 1529–1555, 2021, doi: 10.1214/21-AOAS1439.

[7] J. de la Torre and J. A. Douglas, "Higher-Order Latent Trait Models for Cognitive Diagnosis," *Psychometrika*, vol. 69, no. 3, pp. 333–353, 2025, doi: 10.1007/BF02295640.

[8] P. Zhan, W. Ma, H. Jiao, and S. Ding, "A Sequential Higher Order Latent Structural Model for Hierarchical Attributes in Cognitive Diagnostic Assessments," *Appl. Psychol. Meas.*, vol. 44, no. 1, pp. 65–83, 2020, doi: 10.1177/0146621619832935.

[9] Y. Cui, J. Lu, J. Zhang, N. Shi, J. Liu, and X. Meng, "A stochastic approximation expectation maximization algorithm for estimating Ramsay-curve three-parameter normal ogive model with non-normal latent trait distributions," *Front. Psychol.*, vol. 13, no.1, pp. 1–20, 2022, doi: 10.3389/fpsyg.2022.971126.

[10] T. Y. Kwon, A. C. Huggins-Manley, J. Templin, and M.-Y. Zheng, "Modeling hierarchical attribute structures in diagnostic classification models with multiple attempts," *J. Educ. Meas.*, vol. 61, no. 2, pp. 198–218, 2024, doi: 10.1111/jedm.12387.

[11] X. Li, Y. Chen, H. Yang, and J. Geng, "A Joint Diagnosis Model Using Response Time and Accuracy for Online Learning Assessment," *Electronics*, vol. 14, no. 19, pp. 1–22, 2025, doi: 10.3390/electronics14193873.

[12] J. Guo, Z. Yan, Y. Sato, and Q. Zuo, "Salmon Salar Optimization: A Novel Natural Inspired Metaheuristic Method for Deep-Sea Probe Design for Unconventional Subsea Oil Wells," *J. Mar. Sci. Eng.*, vol. 12, no. 10, pp. 1-12, 2024, doi: 10.3390/jmse12101802.

[13] Y. Hong , "Search-Efficient Computerized Adaptive Testing," in *Proceedings of the 32nd ACM International Conference on Information and Knowledge Managemen, Birmingham: Association for Computing Machinery*, vol. 2023, no. 1, pp. 773–782, 2023. doi: 10.1145/3583780.3615049.

[14] K. Gorney and M. D. Reckase, "Using Multiple Maximum Exposure Rates in Computerized Adaptive Testing," *J. Educ. Meas.*, vol. 62, no. 2, pp. 360–379, 2025, doi: 10.1111/jedm.12436.

[15] Q. Liu , "Survey of Computerized Adaptive Testing: A Machine Learning Perspective," *Preprint arXiv*, vol. 1, no. 1, pp.1-12, 2024, doi: 10.48550/arXiv.2404.00712.

[16] Y. Zhang, D. Wang, X. Gao, Y. Cai, and D. Tu, "Development of a computerized adaptive testing for internet addiction," *Front. Psychol.*, vol. 10, no. May, pp. 1–12, 2019, doi: 10.3389/fpsyg.2019.01010.

[17] B. Klein and K. Kovacs, "The performance of ChatGPT and Bing on a computerized adaptive test of verbal intelligence," *PLoS One*, vol. 19, no. 7 July, pp. 1–9, 2024, doi: 10.1371/journal.pone.0307097.

[18] M. A. Samsudin, T. S. Chut, M. E. Ismail, and N. J. Ahmad, "A calibrated item bank for computerized adaptive testing in measuring science TIMSS performance," *Eurasia J. Math. Sci. Technol. Educ.*, vol. 16, no. 7, pp. 1–15, 2020, doi: 10.29333/EJMSTE/8259.

[19] M. R. Harvey and C. Harrison, "Computerized adaptive testing with the I-HaND scale for monitoring patients with upper limb nerve pathology," *J. Hand Surg. (European* Vol., vol. 50, no. 8, pp. 1033–1040, 2025, doi: 10.1177/17531934251324381.

[20] H. Wang , "GMOCAT: A Graph-Enhanced Multi-Objective Method for Computerized Adaptive Testing," in *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Minin, California: Association for Computing Machinery*, vol. 2023, no. 1, pp. 2279–2289, 2023. doi: 10.1145/3580305.3599367.

[21] Q. Tan, Y. Cai, F. Luo, and T. Dongbo, "Development of a High-Accuracy and Effective Online Calibration Method in CD-CAT Based on Gini Index," *J. Educ. Behav. Stat.*, vol. 48, no. 1, pp. 103–141, 2022, doi: 10.3102/10769986221126741.

[22] M. She, Y. Li, D. Tu, and Y. Cai, "Computerized Adaptive Testing for Sleep Disorders: Development of an Item Bank and Validation in a Simulated Study," *Eur. J. Heal. Psychol.*, vol. 28, no. 3, pp. 89–100, 2021, doi: 10.1027/2512-8442/a000076.

[23] A. D. Wulansari, D. P. Kirana, and R. Mufanti, "Development of a Computerized-Adaptive Test to Measure English Vocabulary Size with IRT," *Indones. J. Learn. Adv. Educ.*, vol. 5, no. 3, pp. 277–294, 2023, doi: 10.23917/ijolae.v5i3.22953.

[24] A. S. Yaro, F. Maly, P. Prazak, and K. Maly, "Outlier Detection Performance of a Modified Z-Score Method in Time-Series RSS Observation With Hybrid Scale Estimators," *IEEE Access*, vol. 12, no. 1, pp. 12785–12796, 2024, doi: 10.1109/ACCESS.2024.3356731.

[25] L. Ciccione, G. Dehaene, and S. Dehaene, "Outlier Detection and Rejection in Scatterplots: Do Outliers Influence Intuitive Statistical Judgments?," *J. Exp. Psychol. Hum. Percept. Perform.*, vol. 49, no. 1, pp. 129–144, 2022, doi: 10.1037/xhp0001065.

[26] A. K. Jadiya, A. Chaudhary, and R. Thakur, "Polymorphic SBD preprocessor: A preprocessing approach for social big data," *Indian J. Comput. Sci. Eng.*, vol. 11, no. 6, pp. 953–961, 2020, doi: 10.21817/indjcse/2020/v11i6/201106169.

[27] E. A. Pambudi, A. G. Fahrezi, and A. Y. Badharudin, "Forecasting Brown Sugar Production Using k-NN Minkowski Distance and Z-Score Normalization," *J. Inf. Syst. Informatics*, vol. 5, no. 2, pp. 580–589, 2023, doi: 10.51519/journalisi.v5i2.485.

[28] H. Henderi, “Comparison of Min-Max normalization and Z-Score Normalization in the K-nearest neighbor (kNN) Algorithm to Test the Accuracy of Types of Breast Cancer,” *IJIIS Int. J. Informatics Inf. Syst*., vol. 4, no. 1, pp. 13–20, 2021, doi: 10.47738/ijiis.v4i1.73.

[29] S. Dai and D. Svetina Valdivia, “Dealing with Missing Responses in Cognitive Diagnostic Modeling,” *Psych*, vol. 4, no. 2, pp. 318–342, 2022, doi: 10.3390/psych4020028.

[30] P. Zhan, F. Li, and H. Jiao, “Editorial: Cognitive Diagnostic Assessment for Learning,” *Front. Psychol.*, vol. 12, no. November, pp. 10–12, 2021, doi: 10.3389/fpsyg.2021.806636.

[31] S. Liu, H. Qian, M. Li, and A. Zhou, “QCCDM: A Q-Augmented Causal Cognitive Diagnosis Model for Student Learning,” *Front. Artif. Intell. Appl.,* vol. 372, no. 1, pp. 1536–1543, 2023, doi: 10.3233/FAIA230434.

[32] C. Ma, J. Ouyang, and G. Xu, “Learning Latent and Hierarchical Structures in Cognitive Diagnosis Models,” *Psychometrika,* vol. 88, no. 1, pp. 175–207, 2023, doi: 10.1007/s11336-022-09867-5.

[33] Y. Gu and G. Xu, “Identifiability of Hierarchical Latent Attribute Models,” *Stat. Sin*., vol. 33, no. 4, pp. 2561–2591, 2023, doi: 10.5705/ss.202021.0350.

[34] X. Shi, X. Ma, W. Du, and X. Gao, “Diagnosing Chinese EFL learners’ writing ability using polytomous cognitive diagnostic models,” *Lang. Test*., vol. 41, no. 1, pp. 109–134, 2024, doi: 10.1177/02655322231162840.

[35] J. Güsten, D. Berron, E. Düzel, and G. Ziegler, “Bayesian modeling of item heterogeneity in dichotomous recognition memory data and prospects for computerized adaptive testing,” *Sci. Rep.*, vol. 12, no. 1, pp. 1–17, 2022, doi: 10.1038/s41598-022-04997-3.

[36] Y. Choi and C. McClenen, “Development of Adaptive Formative Assessment System Using Computerized Adaptive Testing and Dynamic Bayesian Networks,” *Appl. Sci*., vol. 10, no. 22, pp. 1-16, Nov. 2020, doi: 10.3390/app10228196.

[37] K. C. T. Han, “Components of the item selection algorithm in computerized adaptive testing,” *J. Educ. Eval. Health Prof.*, vol. 15, no. 7, pp. 1–13, 2018, doi: 10.3352/jeehp.2018.15.7.

[38] N. van Buuren and T. Eggen, “Latent-Class-Based Item Selection for Computerized Adaptive Progress Tests,” *J. Comput. Adapt. Test.*, vol. 5, no. 2, pp. 22–43, 2017, doi: 10.7333/1704-0502022.

[39] H. A. Ammar, R. Adve, S. Shahbazpanahi, G. Boudreau, and K. V. Srinivas, “POMDP-based Handoffs for User-Centric Cell-Free MIMO Networks,” in P*roceedings - IEEE Global Communications Conference, GLOBECOM, Rio de Janeiro: IEEE*, vol. 2022, no. 1, pp. 341–346, 2022. doi: 10.1109/GLOBECOM48099.2022.10001033.

[40] A. Loisy and R. A. Heinonen, “Deep reinforcement learning for the olfactory search POMDP: a quantitative benchmark,” *Eur. Phys. J. E*, vol. 46, no. 3, pp. 1–14, 2023, doi: 10.1140/epje/s10189-023-00277-8.

[41] M. L. Littman, “A tutorial on partially observable Markov decision processes,” *J. Math. Psychol*., vol. 53, no. 3, pp. 119–125, 2009, doi: 10.1016/j.jmp.2009.01.005.

[42] Y. Wang, A. K. Bozkurt, and M. Pajic, “Reinforcement Learning with Temporal Logic Constraints for Partially-Observable Markov Decision Processes,” *arXiv Cornell Univ*., vol. 2021, no. 1, pp. 1–6, 2021.

[43] M. Zuccotto, M. Piccinelli, A. Castellini, E. Marchesini, and A. Farinelli, “Learning State-Variable Relationships in POMCP: A Framework for Mobile Robots,” *Front. Robot.* AI, vol. 9, no. July, pp. 1–18, 2022, doi: 10.3389/frobt.2022.819107.

[44] J. A. Filar, Z. Qiao, and N. Ye, “POMDPs for Sustainable Fishery Management,” in *23rd International Congress on Modelling and Simulation - Supporting Evidence-Based Decision Making: The Role of Modelling and Simulation, MODSIM, Canberra: The Modelling and Simulation Society of Australia and New Zealand Inc,* vol. 2019, no. 1, pp. 645–651, 2019. doi: 10.36334/modsim.2019.g2.filar.

ISSN 2723-6471

[45] A. A. Sulyok and K. Karacs, "Towards Using Fully Observable Policies for POMDPs," in 2*nd International Conference on Computing and Machine Intelligence (ICMI), Istanbul: IEEE,* vol. 2022, no. 1, pp. 1–5, 2022. doi: 10.1109/ICMI55296.2022.9873768.

[46] F. Wang , "NeuralCD: A General Framework for Cognitive Diagnosis," *IEEE Trans. Knowl. Data Eng*., vol. 35, no. 8, pp. 8312–8327, 2023, doi: 10.1109/TKDE.2022.3201037.

[47] M. D. Reckase, Multidimensional Item Response Theory. London: Springer, 2009. doi: 10.1016/S0169-7161(06)26018-8.